WEST Search History

Hide Items Restore Clear Cancel

DATE: Thursday, November 10, 2005

Hide?	Set Name	Query	Hit Count
		DB=USPT; PLUR=YES; OP=OR	
☐	L45	6078733 and @ad<19971014	1
☐	L44	6078733	11
☐	L43	(5566170 5878225 5930830 5991299 6034963 6061368).pn.	6
☐	L42	alacritech.as.	18
		DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR	
☐	L41	L38 and @ad<1997	0
☐	L40	L39 and @ad<1997	0
☐	L39	L38 and @ad<19971014	0
☐	L38	alacritech.as.	80
☐	L37	(intelligent adj network adj interface) and @ad<19971014	51
☐	L36	L35 and @ad<19971014	150
☐	L35	(intelligent adj network adj interface) inic	5312
☐	L34	(6009478 6034963).pn.	4
☐	L33	(5996024 6226680 6247060).pn.	6
☐	L32	L31 and @ad<19971014	6
☐	L31	(offload$3 off-load$3) with (nic (network adj interface$3)) with (host$3 cpu)	125
☐	L30	L29 and @ad<19971014	0
☐	L29	(tcp ip) with (offload$3 off-load$3) with (nic (network adj interface$3))	138
☐	L28	6427171	41
☐	L27	L26 and offload$3	49
☐	L26	L25 and @ad<19971014	1342
☐	L25	(host cpu) with (nic (network adj interface$3))	9119
☐	L24	L23 and @ad<19971014	1
☐	L23	6247060	64
☐	L22	6247060	64
☐	L21	L20 and 709/$.ccls.	12
☐	L20	L19 and ((bypass$5 offload$5) with (cpu microprocessor$3 processor$5))	72
☐	L19	L18 and @ad<19971014	2189
☐	L18	(microporcessor$5 processor$3 cpu) with (nic (network adj interface$3))	14832
☐	L17	20010025315	2
☐	L16	20010025315	2

☐	L15	5937436	10
☐	L14	L13 and @ad<19971014	2
☐	L13	offload$5 with (microporcessor$5 processor$3 cpu) with (nic (network adj interface$3))	90
☐	L12	L11	13
	DB=USPT; PLUR=YES; OP=OR		
☐	L11	offload$5 with (microporcessor$5 processor$3 cpu) with (nic (network adj interface$3))	13
☐	L10	L8 and @ad<19971014	15
☐	L9	L7 and @ad<19971014	949
☐	L8	L7 and (offload$5 with (cpu processor$3 microporcessor$5))	56
☐	L7	network adj interface adj card$3	4484
☐	L6	L5 and @ad<19971014	2
☐	L5	intelligent adj network adj interface adj card$3	28
☐	L4	6061742.pn.	1
☐	L3	6427171	9
☐	L2	6427171	9
☐	L1	6697868	3

END OF SEARCH HISTORY

Sign in

Web Images Groups News Froogle Local New! more »

"offload cpu " "network interface card" Search Advanced Search Preferences

Web Results **1** - **10** of about **22** for "**offload cpu** " "**network interface card**". (**0.35** seconds)

Gigabit Ethernet: December 1997 Update
... for your servers that support these features and **offload CPU** in the process. ... a PCI and SBus **network interface card**(s); a router interface card(s) or ...
www.corecom.com/html/gigabit.html - 21k - Cached - Similar pages

Instant Rules of Thumb for Tuning and Sizing NT Server
Network Interface Card Settings. Setting the NIC properly is an area that is commonly ... Whenever possible, **offload CPU** operations to secondary devices. ...
www.microsoft.com/technet/ archive/winntas/maintain/instant.mspx - 254k - Cached - Similar pages

Instant Rules of Thumb for Tuning and Sizing NT Server
... **Network Interface Card** Settings. Setting the NIC properly is an area that is commonly taken for granted. ... **Offload CPU** Intensive Operations. ...
www.microsoft.com/technet/ prodtechnol/winntas/maintain/instant.mspx - 101k - Supplemental Result - Cached - Similar pages

[PDF] Windows NT/2000 Performance and Capacity Key Metrics INFORMATION ...
File Format: PDF/Adobe Acrobat - View as HTML
get info concerning the **network interface card**. Select Start/Settings/Control ... **Offload CPU** intensive work. * Remove faulty Hardware. ...
regions.cmg.org/regions/ nycmg/presentations/CMGNTNY01.pdf - Similar pages

EP1359724
... this capability and developed methods to **offload CPU** intensive tasks and ... TCP connection from the core stack to a **network interface card** (NIC). ...
swpat.ffii.org/pikta/txt/ep/1359/724/ - 100k - Cached - Similar pages

[PDF] University of Manitoba
File Format: PDF/Adobe Acrobat - View as HTML
may be embedded in a **network interface card**,NIC,or host bus adapter,HBA. ... TOE not only **offload CPU** processing to increase application perfor- ...
www.phatpackets.com/papers/ PhatpacketsTransportOffloadSiliconPotential-WithAppendices.pdf - Similar pages

[PDF] SNA IDIG
File Format: PDF/Adobe Acrobat - View as HTML
Copyright © 1998-2001 Cisco Systems, Inc. All rights reserved. Important Notices and Privacy Statement. Printed in USA. Catalyst, Cisco, Cisco IOS, ...
www.psionic.com/application/pdf/en/us/ guest/tech/tk658/c1501/ccmigration_09186a008007cd59.pdf - Supplemental Result - Similar pages

Method to offload a network stack patent
... that peripheral device manufacturers tried to do was to offload the entire TCP connection from the core stack to a **network interface card** (NIC). ...
www.freshpatents.com/ Method-to-offload-a-network-stack-dt20050428ptan20050091412.php - 21k - Supplemental Result - Cached - Similar pages

[PDF] Monitoring NT Performance
File Format: PDF/Adobe Acrobat - View as HTML
will get info concerning the **network interface card**. Select start/settings/control ... **offload**

CPU intensive work. Remove faulty hardware. NT service packs ...
www.ohvcmg.org/pdf/CMG02g2.pdf - Similar pages

[PDF] Monitoring NT Performance
File Format: PDF/Adobe Acrobat - View as HTML
will get info concerning the **network interface card**. Select Start/settings/control panel/networks/services/add and ad d the SNMP service. you will ...
www.srmassoc.com/CMG02g2.pdf - Supplemental Result - Similar pages

Gooogle ▶
Result Page: 1 2 3 Next

Free! Get the Google Toolbar. Download Now - About Toolbar

Google ▾ Search ▾ 377 blocked Check ▾ AutoLink ▾ AutoFill

"offload cpu " "network interface card Search

Search within results | Language Tools | Search Tips | Dissatisfied? Help us improve

Google Home - Advertising Programs - Business Solutions - About Google

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: The ACM Digital Library The Guide

+"NIC" +"offload"

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

Published before October 1997
Terms used NIC offload

Found **1** of **88,246**

Sort results by: relevance
Display results: expanded form

Save results to a Binder
Search Tips
Open results in a new window

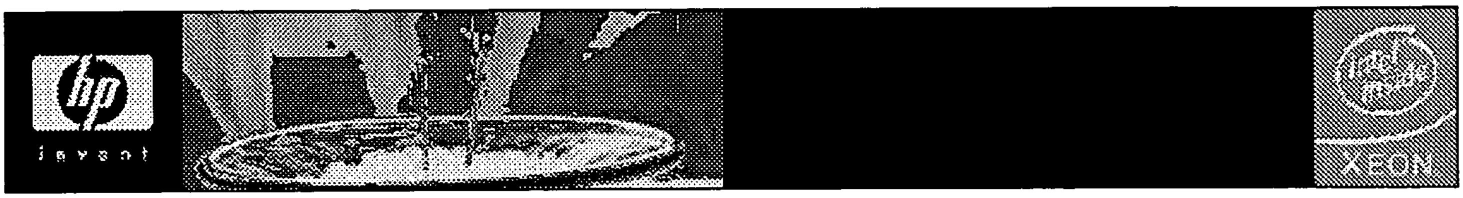
Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 1 of 1

Relevance scale

1 Computer Communication Networks: Approaches, Objectives, and Performance Considerations
Stephen R. Kimbleton, G. Michael Schneider
September 1975 **ACM Computing Surveys (CSUR)**, Volume 7 Issue 3
Publisher: ACM Press
Full text available: pdf(3.99 MB) Additional Information: full citation, references, citings, index terms

Results 1 - 1 of 1



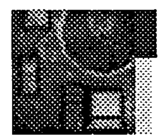
COMPUTERWORLD
SNIA
STORAGE NETWORKING WORLD ONLINE

THE ONLINE STORAGE NETWORKING MAGAZINE FOR IT LEADERS

USER FEEDBACK

Click here to tell us what you thought of the story

WHY IMPLEMENT STORAGE NETWORKS



ARTICLE POSTED April 15th, 2002

Thanks to acceleration technologies, iSCSI is feasible

By Larry Boucher

iSCSI is a protocol that has attracted a great deal of attention in the storage industry. With iSCSI, users will be able to create SANs practically anywhere over standard Ethernet cabling — without requiring dedicated Fibre Channel networks to carry the data between the server and the storage devices. With iSCSI, remote mirroring and backup will become readily available because the distance limitations of Fibre Channel will be irrelevant in a world where data is transferred using standard TCP/IP and Ethernet for transport (see companion story on home page, "iSCSI developments bode well for commercial acceptance).

The emergence of Gigabit Ethernet, and the impending move toward 10 Gigabit Ethernet, will enable the transfer of data using iSCSI protocol at full device speeds. However, to perform these high-speed transfers, a new class of network device will be required.

Until recently, the **CPU** in a server or workstation was responsible for managing the TCP/IP processing necessary for sending and receiving data over a network, with a **network interface card** (**NIC**) providing the actual interface to the network. In the days when CPUs operated at considerably higher speeds than the network data flow, this was not a problem. However, with Gigabit Ethernet, processing of TCP/IP on the **host CPU** has often required as much as 100% of the processor's cycles.

With the addition of iSCSI processing on the **host CPU**, the result will be 100% **CPU** utilization — in effect, locking the computer for any other processes and significantly reducing network performance, because even

the fastest **CPU** will be unable to process TCP/IP and iSCSI at full wire speed.

Offloading server processing
The solution to this problem involves the use of a "TCP/IP **offload** engine" (TOE) that offloads some or all of the TCP/IP processing from the server. In essence, a TOE applies the task of TCP/IP processing to one or more chips that are optimized for this task.

With the addition of iSCSI, further processing is required. As with TCP/IP, the iSCSI processing can be done in software on the server **CPU**, or it can be offloaded to a device specially designed to perform the necessary processing. In all cases, however, to achieve the necessary performance for high-speed networking and/or storage processing, the most effective approach is to **offload** the processing from the **host CPU** to a specialized device.

Three iSCSI implementation approaches
Three basic approaches are being taken to implement iSCSI connectivity.

Standard NICs
Early implementations of iSCSI were proposed using standard Gigabit NICs that left the processing of iSCSI protocol to the **host CPU**. This approach may be the fastest to market because it simply incorporates standard components. But this simple approach is not viable because it requires considerable **CPU** resources. In most cases, the **host CPU** will run at 100% simply to process TCP/IP and iSCSI, and no resources will be available for other computing processes.

Even assuming 100% utilization of a Gigahertz processor, protocol processing with this approach will likely not keep up with wire-speed data transfer, especially when more than one network port or **NIC** is coupled to the **CPU**. Thus, although this approach supports Ethernet failover, load balancing and link aggregation, its poor performance makes it unacceptable for nearly all users.

Full-offload iSCSI HBAs
Taking a cue from the Fibre Channel world, a number of vendors are building iSCSI **host** bus adapters (HBAs). This new class of product uses a full-**offload** approach for iSCSI and TCP/IP protocol processing.

Full-**offload** adapters remove the entire TCP/IP stack from the **host CPU**. As a result, the **host** and network do not see the iSCSI adapters as **NIC** controllers. These cards are not designed to handle generalized networking tasks. For instance, the method for performing failover and link aggregation will not be able to follow a standard Ethernet implementation. Full-**offload** cards may use traditional storage failover facilities such as Active-Active designed for Fibre Channel.

For an iSCSI HBA, TCP/IP and iSCSI protocol processing may be performed on an embedded processor with either a standard Gigabit Ethernet controller or a specially designed application-specific integrated circuit (ASIC). Intel's recently announced PRO/1000 T IP Storage Adapter uses an Intel 80200 (XScale) processor with a standard GbE controller to perform the TCP/IP and iSCSI processing tasks.

Intel claims to have been able to transfer data over iSCSI at speeds ranging from 37.5M bps to 87.5M bps, with 3% to 5% **CPU** usage. However, without data on how the test was run, it is difficult to accurately compare this card's performance with that of other products.

Adaptec is attempting an ASIC implementation, supplementing the Intel 80200 processor with its Storage Protocol Accelerator (SPA) — AIC-7211 ASIC in the ASA-7211 iSCSI Adapter. Like the Intel card, the Adaptec iSCSI HBA uses an Xscale processor. At this time, however, the company has not provided any specifics relating to data throughput or the amount of **CPU** cycles required during data transmissions.

Data-path offload IS-NIC
The term "integrated storage **network interface card**" (IS-**NIC**) refers to an adapter that can be used as a network interface for moving Ethernet file data, as a storage interface for moving iSCSI block data, and as a dual-purpose card functioning as a **NIC** and as an iSCSI device simultaneously. This approach offers the advantages of both the iSCSI HBA and the standard **NIC**, while offering none of the disadvantages associated with isolating these functions.

Rather than move the entire protocol stack onto the adapter, an IS-**NIC** uses a technique called data-path **offload**; namely, data movement is performed in a custom-designed ASIC, and the TCP connection and error management is handled in **host** software. Although the **host CPU** may rarely have to control the data path, it is advantageous for the **host CPU** to be able to take control of the data path under certain conditions, such as error management.

Being able to control data flow enables the **host** computer to perform failover, load balancing and link aggregation using Ethernet standard protocols. If the **host CPU** or the network switch detects that one iSCSI card or link is handling a disproportionate amount of data, it can redistribute the data to other iSCSI cards.

The data-path **offload** approach eliminates the need to design a large, complex chip for rare error occurrences and enables the smallest footprint and the most power-efficient design for TCP/IP and iSCSI **offload**.

The Alacritech 1000x1 Single-Port Server and Storage Accelerator, which began shipping in October 2001, implements data-path **offload** for TCP/IP and iSCSI traffic. The data-path **offload** solution provides comparable or better performance and efficiency, compared with any Fibre Channel or iSCSI HBA. Alacritech recently reported successful wire-speed transfers of 219.64M bps with less than 8% **CPU** utilization for this Gigabit Ethernet IS-**NIC**. Testing was performed with a Nishan IP storage switch that converted iSCSI into Fibre Channel and moved the data to a Hitachi Data Systems SAN device.

An evolving standard
Although iSCSI adapters are currently available, the iSCSI specification has not yet been approved by the Internet Engineering Task Force (IETF). Approval is expected by mid-2002.

Because the specification may undergo additional modifications before an approved spec is agreed upon, it will be impossible to guarantee absolute adherence to the spec prior to its release. However, it is very safe to assume that vendors that are developing iSCSI cards are working hard to assure compatibility, irrespective of the final completion of the spec.

What the future holds
iSCSI is a logical step forward in the control of data devices. It eliminates the need for specialized HBAs in servers and workstations and allows block-level data to be transferred over a standard Ethernet network.

It will be interesting to see how the industry reacts to the various iSCSI alternatives — the processor-based iSCSI adapter that performs full **offload**, the ASIC-based card that performs full **offload**, and a programmable ASIC device that performs data-path **offload**. What should be clear, however, is that the industry is already responding with iSCSI devices, and that iSCSI is gaining momentum as a significant storage protocol.

In this article, we've looked at the three basic methods for implementing iSCSI as an initiator on the server side. Vendors are working on iSCSI-to-Fibre Channel gateways that provide an iSCSI interface to present Fibre Channel SANs. Although the number of iSCSI products currently available is small, it is clear that many vendors are hard at work developing highly compatible iSCSI products.

Larry Boucher is the Chairman and CEO of Alacritech, Inc.

Groups Home | Help | Sign in

Web Images **Groups** News Froogle Local New! **more »**

"network interface" "host" "offload" Search Advanced Groups Search Preferences

1 Jan 1981 – 14 Oct 1997

Members: Sign in
New users: Join

Recently visited [clear]
comp.os.ms-w...g.tcp-ip
comp.protocols.tcp-ip
comp.unix.questions

Groups Alerts

Create a new group
About Google Groups

Searched all groups

Results **1 - 9** of 9 from **Jan 1, 1981** to **Oct 14, 1997** for **"network interface" "host" "offload"** (**0.09** seconds)

Sorted by relevance Sort by date

"**host**-based" vs. "front-end-processor" PC network interfaces
... and acknowledgement o partially **offload** keystroke processing ... Separation between the
host and onboard portions is ... The EXOS205 **Network Interface** Card for PC's is ...

comp.protocols.tcp-ip.ibmpc - May 12 1988, 8:34 pm by Chuck Kollars - 3 messages - 3 authors

multiple IP addrs on single net interface
... This **host** has a single **network interface**. ... HP gotcha #2: The code will almost certainly
not work on interfaces using checksum **offload** (eg. FDDI). ...
comp.unix.questions - Mar 11 1995, 5:14 am by Steinar Haug - 4 messages - 4 authors

Shared memory comm. on HP machines, possible?
... go to that hosts primary "real" **network interface** (such as lan0 ... not think that there
is checksum **offload**, and will ... If instead, you add some **host** routes for your ...
comp.parallel.pvm - Jun 23 1994, 7:52 pm by Rick Jones - 6 messages - 6 authors

DISPLAY=:0.0
... X calls go thru **network interface** if "**host**:0" is ... will go part-way into the **network interface's** _driver_ but ... to have a larger MTU and provides checksum **offload**. ...
comp.sys.hp.hpux - Nov 8 1996, 5:15 pm by Rick Jones - 19 messages - 14 authors

Fragen zu Protokollen
... Processing Environment" protocol, may be employed to **offload** Network and ... be erected
on an ARM LI - L II (**Network Interface** and **Host-Host** Layers) "foundation ...
de.comp.standards - Nov 15 1994, 6:41 pm by Kai Henningsen - 9 messages - 7 authors

Linux/tcpdump/NIC cards (traffic analysis)
... want a :"smarter" nic card that will **offload** some of the ... gives a breakdown of traffic
by **host** Let me ... does it use to put the **network interface** into promiscuous ...
comp.protocols.tcp-ip - Oct 5 1997, 7:24 am by Aaron Newsome - 7 messages - 7 authors

new Inmos Transputer (was Re: 88000 Question)
So you **offload** the protocol processing to a dedicated ... 100Mbps to the general purpose
host running Unix? ... because a transputer doesn't have a **network interface**. ...
comp.sys.next.misc - Jul 29 1991, 2:56 am by Scott Hess - 57 messages - 29 authors

Introduction to TCP/IP Administration by Chuck Hendrick
... the machine has more than one **network interface**, you must ... point lines or switched

lines directly to each **host**. ... It often makes sense to **offload** the routing and ...
comp.os.ms-windows.networking.tcp-ip - Dec 1 1995, 5:37 am by Greg E Hersh - 1 message - 1 author

Embedded Processor and Microcontroller primer and FAQ
... a dedicated [co]processor to **offload** some of ... for another keystroke, or information
from the **host**. ... multi-master, multi-slave **network interface** with collision ...
comp.robotics.misc - Oct 1 1996, 7:23 pm by Russ Hersch - 1 message - 1 author

In order to show you the most relevant results, we have omitted some entries very similar to the 9 already displayed.
If you like, you can repeat the search with the omitted results included.

Search